

## Research Article

# Grey Wolf Resampling-Based Rao-Blackwellized Particle Filter for Mobile Robot Simultaneous Localization and Mapping

Yong Dai <sup>1</sup> and Ming Zhao <sup>2</sup>

<sup>1</sup>*School of Automation and Electrical Engineering, Shenyang Ligong University, Shenyang 110159, China*

<sup>2</sup>*School of Applied Technology, University of Science and Technology Liaoning, Anshan 114051, China*

Correspondence should be addressed to Ming Zhao; [as\\_zmyli@163.com](mailto:as_zmyli@163.com)

Received 19 September 2021; Accepted 7 October 2021; Published 26 October 2021

Academic Editor: L. Fortuna

Copyright © 2021 Yong Dai and Ming Zhao. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An artificial intelligent grey wolf optimizer (GWO)-assisted resampling scheme is applied to the Rao-Blackwellized particle filter (RBPF) in the simultaneous localization and mapping (SLAM). By doing this, we can make the diversity of the particles resampling and then obtain a better localization accuracy and fast convergence to realize indoor mobile robot SLAM. In addition, we propose an adaptive local data association (Range-SLAM) scheme to improve the computational efficiency for the algorithm of the nearest neighbor (NN) data association in the iteration of the RBPF prediction. Through the experiment and simulations, the proposed SLAM schemes have fast convergence, accuracy, and heuristics. Therefore, the improved RBPF and new data association schemes presented in this paper can provide a feasible method for the indoor mobile robot SLAM.

## 1. Introduction

In a completely unknown environment, the mobile robots just relying on their carrying sensors to get environment information and successfully make simultaneous localization and mapping (SLAM) is a basic step to realize autonomous intelligent navigation [1]. Meanwhile, it is a strategy that could be useful in a wider area of research, such as the motion control agents and microfluidics [2, 3]. Extended Kalman filter (EKF) is the most popular method used in the SLAM [4, 5]. However, the SLAM based on the EKF cannot create a large map and it also has a problem of the data association computation efficiency. Therefore, in recent years, the RBPF [6, 7] has been introduced as an effective method to solve the SLAM problem. The core idea of the RBPF technique is to use a particle filter (PF) and the Kalman filter (KF) to predict features in the map, where each particle represents a robot position and attitude. But there exists a new problem; that is, the particles have difficulty in the proposal distribution selection and resampling. In the SLAM problem, the experience tells us that we can select the robot motion model as the proposal distribution, but the

method of PF resampling is still less efficient and accurate. Unfortunately, the resampling technique is more important, which can not only avoid particles degeneracy but also determine the robot localization accuracy. The traditional resampling method only pays attention to the identical distribution (ID) attribute [8] of the particle in the posterior probability prediction but ignores the particle diversity, which leads to poor robot positioning prediction accuracy [9].

The contribution of this paper lies in the following: (1) introducing an artificial intelligent grey wolf optimization (GWO)-assisted resampling scheme for making the diversity of particles and (2) proposing an adaptive local data association (Range-SLAM) for NN's efficiency. The above two schemes solve the most intractable problem of conventional RBPF for SLAM. In our scheme of resampling proposed, it can not only guarantee the ID property but also not lose the diversity of particles. In addition, the proposed new data association method improves the computing efficiency in the RBPF. Therefore, the improved RBPF becomes more effective and flexible in the SLAM.

The organization of this paper is as follows: after reviewing the normal RBPF, we explain the Rao-Blackwellized resampling idea in Section 3. Sections 4 and 5 introduce the exploration techniques of GWO-assisted resampling. Section 6 describes a new data association method. Finally, Section 7 contains the experimental results carried out on real robots as well as in the simulations. Section 8 concludes this study.

## 2. Rao-Blackwellized Filter Review

According to Murphy [10], the key idea of solving the SLAM problem with a Rao-Blackwellized particle filter (RBPF) is to estimate a posterior  $p(X_t|Y_t)$  and divide the state vector  $X_t$  into two parts: one part is  $x_t^p$ , which is estimated using the PF, and the other part is  $x_t^k$ , where Kalman filters (KFs) are used. Denoting the measurements and states up to time  $t$  with  $Y_t = \{y_j\}_{j=0}^t$  and  $X_t = \{x_j\}_{j=0}^t$ , respectively, the joint probability density function (PDF) is given by Bayes' rule as

$$p(x_t^p, x_t^k | Y_t) = \underbrace{p(x_t^k | x_t^p, Y_t)}_{\text{KF}} \underbrace{p(x_t^p | Y_t)}_{\text{PF}}. \quad (1)$$

If the first term  $p(x_t^k | x_t^p, Y_t)$  is linear, it can be optimally estimated using a KF. To obtain the second term  $p(x_t^p | Y_t)$ , it is necessary to apply nonlinear filtering techniques (the PF will be used). It is clear that the PF can be sampled in a lower dimensional state space (e.g., 3 dimensions), when  $x_t^p$  represents the pose of mobile robot. Then the PF and KF can estimate obstacles in map with a known mobile robot pose.

For a mobile robot, the general movement and observation equations can be defined by

$$\begin{cases} X_{t|t-1} = f(X_{t-1|t-1}, u_t) + \varepsilon_t \longrightarrow p(X_{t|t-1} | X_{t-1|t-1}), \\ y_t = h(X_{t|t-1}) + \eta_t \longrightarrow p(y_t | X_{t|t-1}), \end{cases} \quad (2)$$

where  $u_t$  is the control signal;  $\varepsilon_t$  and  $\eta_t$  represent the Gaussian white noise with 0 mean value and their covariances are  $Q_t$  and  $R_t$ .

In the estimation of RBPF-SLAM algorithm, a probability ranging motion model is usually used as the recommended distribution:

$$X_{t|t-1} \sim q(x_t | x_{t-1}, Y_t) = p(X_{t|t-1} | X_{t-1|t-1}), \quad (3)$$

where the proposal distribution probability density function  $q$  is a sampling density function. Then, the importance weight is calculated to compensate for the difference between the proposal distribution (Sequential Importance Sample, SIS) and the target distribution for the pose of the robot by

$$\begin{aligned} \omega_t^{(i)} &\propto \omega_{t-1}^{(i)} \frac{p(y_t | x_t) p(x_t | x_{t-1})}{q(x_t | x_{t-1}, Y_t)} \\ &= \omega_{t-1}^{(i)} p(y_t | x_{t|t}^{k(i)}, x_{t|t}^{p(i)}, Y_{t-1}), \end{aligned} \quad (4)$$

which is proportional to a measurement likelihood. Then, the obtained weights are normalized as follows:

$$\omega_{t+1}^i = \frac{\omega_{t+1}^i}{\sum_{j=1}^N \omega_{t+1}^j}. \quad (5)$$

For dividing the state vector  $X_t$  into two parts, the model of the state can be decomposed into linear/nonlinear Gaussian system [11, 12] as follows:

$$\begin{cases} X_{t+1}^p = f^p(x_t^p) + F^p(x_t^p)x_t^k + G^p(X_t^p)\omega_t^p, \\ X_{t+1}^k = f^k(x_t^p) + F^k(x_t^p)x_t^k + G^k(X_t^p)\omega_t^k, \\ y_t = h(x_t^p) + H^k(x_t^p)x_t^k + e_t, \end{cases} \quad (6)$$

where the state noise is assumed to be white and Gaussian distributed with

$$\begin{aligned} \omega_t &= \begin{bmatrix} \omega_t^p \\ \omega_t^k \end{bmatrix} \sim N(0, Q_t), \\ Q_t &= \begin{bmatrix} Q_t^p & Q_t^{pk} \\ (Q_t^{pk})^T & Q_t^k \end{bmatrix}. \end{aligned} \quad (7)$$

The measurement noise is assumed to be white and Gaussian distributed according to

$$e_t \sim N(0, R_t). \quad (8)$$

Furthermore,  $x_0^p$  is Gaussian:

$$x_0^p \sim N(\bar{x}_0, \bar{P}_0). \quad (9)$$

The density of  $x_0^k$  can be arbitrary, but it is assumed to be known. The Rao-Blackwellization for filtering algorithm is given in Algorithm 1. For the system equation (6), see (7)–(9) for system properties.

The resampling method (Sample Importance Resampling, SIR) is [9] to randomly get rid of unimportant particles and replace them with more likely ones. Generate a uniform distribution of random numbers. Let  $\tilde{\mu} \sim U(0, N^{-1})$ ,  $sw = 0$ , and  $j = 1$ , where  $sw$  is the particle accumulation and the initial value is zero. Detailed algorithm is given in Algorithm 2.

## 3. Resampling (SIR) Analysis

With the particle filter, if the described RBPF runs with exactly the steps described above, it will end up with all the particle weight in one single value. This will degrade estimation performance, because the normal resampling method above is just sure of ID for resampling (details below) and not sure of diversity of particles. Actually, there are two important factors that affect PF estimation performance: one is the proposal distribution which has been solved by motion model  $p(X_{t|t-1} | X_{t-1|t-1})$ , and the other one is the resampling; let us talk about it in detail.

Why resampling? The motivation of resampling includes, but is not limited to, combating sample degeneracy (while avoiding impoverishment) and adjusting the number of particles online, essentially reducing the probability that the filter loses tracking and localization. However, notice

**Step 1:** initialization. For  $i = 1, 2, \dots, N$ ,  $x_{0|0}^{p(i)} \sim p_{x_0}^{p(i)}(x_0^p)$  and set  $\{x_{0|0}^{k(i)}, p_{0|0}^{(i)}\} = \{x_0^k, p_0\}$ , set  $t = 0$ .

**Step 2:** PF measurement update. For  $i = 1, 2, \dots, N$ ,  $\tilde{\omega}_t^{(i)} = p(y_t | x_{t|t}^{k(i)}, x_{t|t}^{p(i)}, Y_{t-1})$  and normalize  $\omega_t^{(i)} = \tilde{\omega}_t^{(i)} / \sum_j \tilde{\omega}_t^{(j)}$ .

**Step 3:** resample  $N$  particles with replacement.  $[\{x_{t|t}^{p(j)}, \omega_t^{(j)}\}_{j=1}^N] = \text{Resample}[\{x_{t|t}^{p(i)}, \omega_t^{(i)}\}_{i=1}^N]$ .

**Step 4:** PF time update and KF.

(a) KF measurement update:  
 $x_{t|t}^{k(i)} = x_{t|t-1}^{k(i)} + K_t^i (y_t - h_i^{(i)} - H_t^{y(i)} x_{t|t-1}^{k(i)})$   
 $P_{t|t}^{(i)} = P_{t|t-1}^{(i)} + K_t^{(i)} M_t^{(i)} K_t^{(i)T}$   
 $M_t^{(i)} = H_t^{y(i)} P_{t|t-1}^{(i)} H_t^{y(i)T} + R$   
 $K_t^{(i)} = P_{t|t-1}^{(i)} H_t^{y(i)T} M_t^{(i)-1}$

(b) PF time update. For  $i = 1, 2, \dots, N$ , predict new practices  $x_{t|t-1}^{p(i)} \sim p(x_{t|t-1}^p | X_t^{p(i)}, Y_t)$ .

(c) PF time update. Set  $z_t^{(i)} = x_{t-1}^{p(i)} - f_t^{p(i)}$ .  
 $X_{t+1|t}^{k(i)} = f_t^{k(i)} + F_t^{k(i)} x_{t|t}^{k(i)} + L_t^{(i)} (z_t^{(i)} - F_t^{p(i)} x_{t|t}^{k(i)})$   
 $P_{t+1|t}^{(i)} = F_t^{k(i)} P_{t|t}^{(i)} F_t^{k(i)T} + G_t^{k(i)} Q^k G_t^{k(i)T} - L_t^{(i)} N_t^{(i)} L_t^{(i)T}$   
 $N_t^{(i)} = F_t^{p(i)} P_{t|t}^{(i)} F_t^{p(i)T} + G_t^{p(i)} Q^p G_t^{p(i)T}$   
 $L_t^{(i)} = F_t^{k(i)} P_{t|t}^{(i)} F_t^{k(i)T} N_t^{(i)-1}$

**Step 5:** increase time and repeat from step 2.

ALGORITHM 1: The Rao-Blackwellization for filtering algorithm.

```

 $[\{x_{t|t}^{p(j)}, \omega_t^{(j)}\}_{j=1}^N] = \text{Resample}[\{x_{t|t}^{p(i)}, \omega_t^{(i)}\}_{i=1}^N]$ 
 $N_{\text{ESS},k} = (\sum_{i=1}^{N_k} (\omega_{t|t}^{(i)})^2)^{-1}$ 
IF  $1 \leq N_{\text{ESS},k} \leq N_k$ , (PF resampling switches between SIR and SIS)
  FOR  $i = 1 : N$ 
     $sw = sw + \omega_t^{(i)}$ 
     $num = 0$  (the number of the  $i$ -th particle copied times and the initial value is zero.)
    WHILE  $sw > \bar{\mu}$ 
       $num = num + 1$ 
       $\bar{\mu} = \bar{\mu} + N^{-1}$ 
       $x_{t|t}^{p(j)} = x_{t|t}^{p(i)}; \omega_t^{(j)} = 1/N; j = j + 1$ 
    END WHILE
  END FOR
END IF

```

ALGORITHM 2: The resampling algorithm.

that resampling basic principle is abided by ID for resampling as well as the diversity of particles [8, 9].

Now a consistent ID metric and the sampling variance are introduced for assessment of the ID attribute of resampling. Discrepancies metric between the numbers of times that the particles are resampled is

$$SV = \frac{1}{M} \sum_{m=1}^M (N_t^{(m)} - N\omega_t^{(m)})^2, \quad (10)$$

where  $N$  is the size after resampling,  $M$  is the size before resampling,  $N_t^{(m)} = \text{floor}(N\omega_t^{(m)})$ , and  $\text{Floor}(\cdot)$  gives the largest integer not exceeding the content. The smaller the SV is, the better the ID quality of the resampling method is. If and only if the two distributions are exactly the same, the SV given in equation (10) is essentially a cost function that provides an efficient metric to measure the discrepancy between the discrete distributions of weighted particles before and after resampling.

From resampling analysis above, the problem of particle weight degeneracy and impoverishment is conventional PF associated and difficult to completely remove. Particle weights will be with serious degradation after using normal resampling (although with a perfect ID), and then particle impoverishment will also be more serious [8], so preventing and solving the dual problem are expected.

In order to overcome the problems caused by resampling, this paper designs a kind of intelligent resampling system; namely, in the sampling space, by means of searching optimal algorithm [13], we search particles with the approximate weight of the highest weight of the particle (but not the same as it), to replace the low weight of the particles in degeneracy. It is not like the normal resampling method above to just copy high weight particles (particles are the same). This will not only ensure the diversity of particles but also ensure that all of the weights of the particles have a high probability with SLAM iterations. Now let us firstly introduce a searching optimization algorithm, that is, GWO.

## 4. Grey Wolf Optimization

GWO is a new kind of metaheuristic biological intelligence algorithm proposed by Mirjalili et al. [14]. The process of imitating the hierarchical leadership and hunting mechanism of wolves during natural wolf hunting is shown in Figure 1. For detailed algorithm, we refer the reader to Seydali's work.

## 5. Grey Wolf Optimization-Assisted Resampling

When resampling, we must keep particles away from impoverishment and invariability. If the resampling has to be taken when particles get the degeneracy, we give up the least weight particles, instead of just mechanically copying the large weight particle and giving a  $1/N$  weight to each one; we do this as follows.

Firstly, we give GWO a searching maximum objective function  $F$  as follows:

$$F = \|x_{t_{HW}}^p - x_t^p\| + a \cdot e^{-|p(Y_t, |x_{t_{HW}}^p) - p(Y_t, |x_t^p)|}, \quad (11)$$

where  $a$  is a constant,  $x_{t_{HW}}^p$  is the particle with highest weight, and  $Y_t$  is a known vector that comes from the observation. In the searching algorithm, parameters  $x_t^p$  to be optimized should obtain a clearly defined scope. In this paper, the wolf algorithm searching range is selecting the maximum weight of the particle to be the center and extending to two sides:

$$X_t^p = \begin{cases} x_t^p: [-\lambda + x_{t_{HW}}^p & \lambda + x_{t_{HW}}^p], \\ y_t^p: [-\lambda + y_{t_{HW}}^p & \lambda + y_{t_{HW}}^p], \\ \theta: [-\pi & \pi], \end{cases} \quad (12)$$

where  $\lambda$  is an arbitrary constant, which can be used to adjust to enlarging and narrowing the range of variation of particles. Above all, we can find that GWO resampling can keep particles away from impoverishment and invariability, instead of just mechanically copying the large weight particle and giving a  $1/N$  weight to each one.

Of course, for the SLAM problem, we should know, in  $F$  function, that  $Y_t$  that comes from the observation should be obtained from data association, and it is also an unavoidable SLAM problem. Let us talk about it in the next section.

## 6. Data Association and its Efficiency Analysis

SLAM refers to a mobile robot in an unknown environment gradually establishing a map of the surrounding environment and determining its own posture. In the SLAM process, the data association is a key issue for real-time map estimation and update based on observational information.

If the valid sensor observations correspond to landmarks presented in the map, they can all form a pair of associations to provide new information for SLAM update; if not, sensor observations will be new landmarks stored in the map. There are a lot of data association methods, but the NN data association algorithm based on Mahalanobis distance has been

widely used in SLAM [15, 16]. In this paper, we will still use NN data association but improve it. The NN algorithm is described as follows. Perform  $t + 1$  time observation and the NN data association in detail.

Assume that the sensor returns  $n$  observations  $z = \{z_1, z_2, \dots, z_n\}$ . For  $n$  observations, we compute Mahalanobis distance  $D_{ni}$  between each  $z_n$  and each stored landmark  $LM_i$ ,  $i = \{1, 2, \dots, M\}$  in the map, and  $D$  is the minimum of  $D_{ni}$ :

$$\begin{cases} D_{ni} = (Z_n - Zp)^T S_{t+1|t}^{-1} (Z_n - Zp), \\ D = \min(D_{ni}); \\ i = 1, 2, \dots, M, \end{cases} \quad (13)$$

where  $Zp = h(X_{t+1|t}^p, X_{t+1|t}^{LM_i})$  is the expected observation of  $i$ -th landmark that existed in map. The innovation covariance matrix for the expected observation is

$$S_{t+1|t} = H_{t+1|t} P_{t+1|t} H_{t+1|t}^T + R_{t+1|t}, \quad (14)$$

where  $H_{t+1|t} = (\partial Z p / \partial X_{t+1|t})$  is the Jacobian matrix of the system of observation equations. The general assumption is  $D_{ni} \sim \chi^2(2)$ . Confidence is given to determine the correlation threshold  $G1$  and augmented threshold  $G2$ ; if  $D_{ni} < G1$ , then  $LM_i$  and  $Z_n$  are a set of acceptable association; if there are sets of acceptable association with  $Z_n$ , then select the landmark with minimum norm distance (ND) to be a set of acceptable associations with  $Z_n$ , where  $ND = D + \log(\|S\|)$ ; if  $D_{ni} > G2$ , then  $Z_n$  is a new landmark to be stored; if  $G1 < D_{ni} < G2$ , then give up  $Z_n$ , and regard it as noises.

**6.1. Efficiency Analysis.** Assume there are  $M$  landmarks in the map right now; from the NN data association analysis, we can see the following:

- (1) If there is a new observed landmark by sensor and data association is made in the map, then it will need to compute  $n * M$  times with NN to solve the SLAM problem. With the increment of the number of observed new landmarks,  $M$  will be expanded largely, and computation is increased.
- (2) For PF in RBPF, its dimension of the system state vector is 3, but, unfortunately, the dimension of the system state vector of KF is  $m$ , where  $m$  represents the existing landmarks in the map right now. In large and complex environment, with the increment of the number of observed new landmarks, the dimension of the system state vector of KF will also increase. Computation of covariance matrix and other matrices will be dramatically increased.

To avoid reducing the operating efficiency and localization inaccuracy, we propose a SLAM algorithm with adaptive observation range (Range-SLAM). The principle of the algorithm is to use a circular local map to estimate the position and attitude of mobile robot in the world coordinate system, while updating the local map, as shown in Figure 2(a).

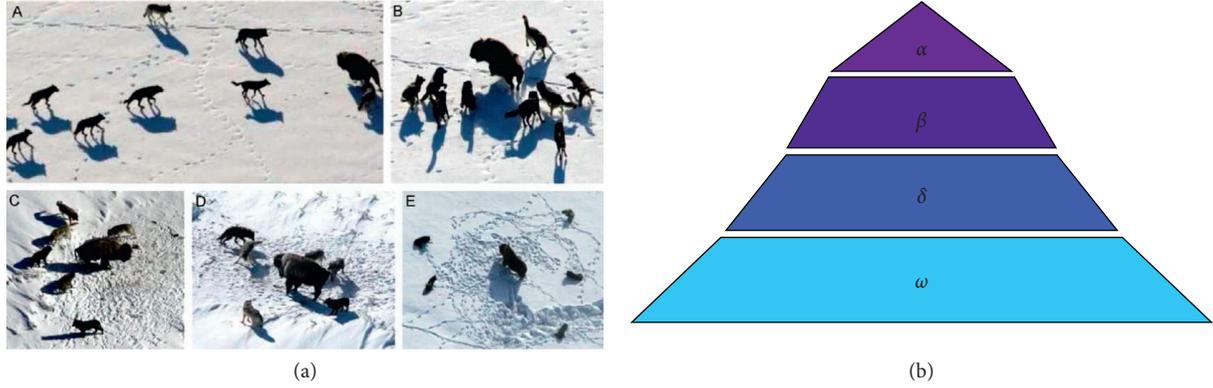


FIGURE 1: (a) Hunting behavior of grey wolves: (A) chasing, approaching, and tracking prey; (B–D) pursuing, harassing, and encircling; (E) stationary situation and attack. (b) Hierarchy of grey wolf (dominance decreases from top down).

In Figure 2,  $S$  represents circular local map;  $S1$  represents observation range;  $S2$  represents constraint area; black points represent landmarks. First, we need to detect whether there are new landmarks in area  $S1$  by constraint conditions:

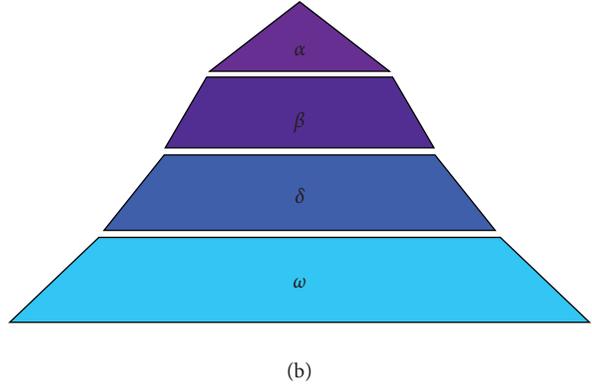
$$\sqrt{(x_{\text{real}_k} - m_j^x)^2 + (y_{\text{real}_k} - m_j^y)^2} < R, \quad (15)$$

where  $\text{Robot}_{\text{real-position}} = [x_{\text{real}_k} \ y_{\text{real}_k}]^T$  is the robot real position in map.  $m_j = [m_j^x \ m_j^y]^T$  is the landmark in observation range;  $R$  is an adaptive radius scalar. If it is a new landmark, then add it to the system state vector matrix. At the same time, for the number of landmarks  $\text{NUM}$  in area  $S$ , if  $\text{NUM} > \text{NUM}_{\text{max}}$ , then reduce the radius  $R$  of circle  $S$ ; otherwise, if  $\text{NUM} < \text{NUM}_{\text{min}}$ , then increase the radius under the premise of  $R_{\text{min}} \leq R \leq R_{\text{max}}$ .

For this purpose, we just do the data association in  $S$  space and not in the entire map space of the landmarks with one to one association; at the same time, the KF state vector is also limited to the dimension of the  $S$  space. This will greatly reduce the computational complexity of the data association, while a more important thing is to reduce the complexity of the KF in the process of computing. It is just like the memory of a computer;  $S$  space only works as the current operation of the domain, and the entire map is like the hard drive; the robot will store and refresh  $S$  space landmarks in the process of detecting new landmarks. The proposed method is easy but a valid one; we can finally determine mobile robot position and attitude accurately by RBPF above.

## 7. Experiment

In this paper, we take two wheels' differential driving mobile robot as an experiment. Its motion model is shown in Figure 3. The state of mobile robot of world coordinate system is represented by the position and orientation vector  $X_k = [x_k, y_k, \theta_k]^T$ , where  $X_k = [x_k, y_k]^T$  are coordinates of mobile robot in  $k$  moment;  $\theta_k$  is the angle between the forward direction of mobile robot and the positive direction of  $x$ -axis of the world coordinate system.



The linear velocity  $v_k$  and rotational angular velocity  $\omega_k$  of mobile robot in  $k$  moment are

$$\begin{cases} v_k = \frac{(v_{R,k} + v_{L,k})}{2}, \\ \omega_k = \frac{(v_{R,k} - v_{L,k})}{L}, \end{cases} \quad (16)$$

where  $v_{R,k}, v_{L,k}$  are the linear velocities of the right and left wheels;  $L$ -axis is the spacing between two wheels. Assuming that the speed of the mobile robot is uniformly changed from  $k-1$  moment to  $k$  moment, the kinematic model of the mobile robot is

$$X_k = \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} x_{k-1} + \left[ \frac{(v_{k-1} + v_k)}{2} \right] \cdot \Delta t \cdot \cos(\theta_{k-1}) \\ y_{k-1} + \left[ \frac{(v_{k-1} + v_k)}{2} \right] \cdot \Delta t \cdot \sin(\theta_{k-1}) \\ \theta_{k-1} + \left[ \frac{(\omega_{k-1} + \omega_k)}{2} \right] \cdot \Delta t \end{bmatrix}, \quad (17)$$

where  $\Delta t$  is the sampling period.

For the state observation model, in this paper, we set that there are totally  $N$  landmarks and the coordinates  $[m_i^x, m_i^y]^T$  ( $i = 1, 2, \dots, N$ ) do not change; then the observation model of mobile robot is

$$Z_k^i = \begin{bmatrix} r_k^i \\ \phi_k^i \end{bmatrix} = \begin{bmatrix} \sqrt{(x_k - m_i^x)^2 + (y_k - m_i^y)^2} \\ \arctan\left(\frac{m_i^y - y_k}{m_i^x - x_k}\right) - \theta_k \end{bmatrix}, \quad (18)$$

where  $r_k^i$  is the distance between the mobile robot and landmark  $i$ ;  $\phi_k^i$  is the angle between the forward direction and connection line of mobile robot.

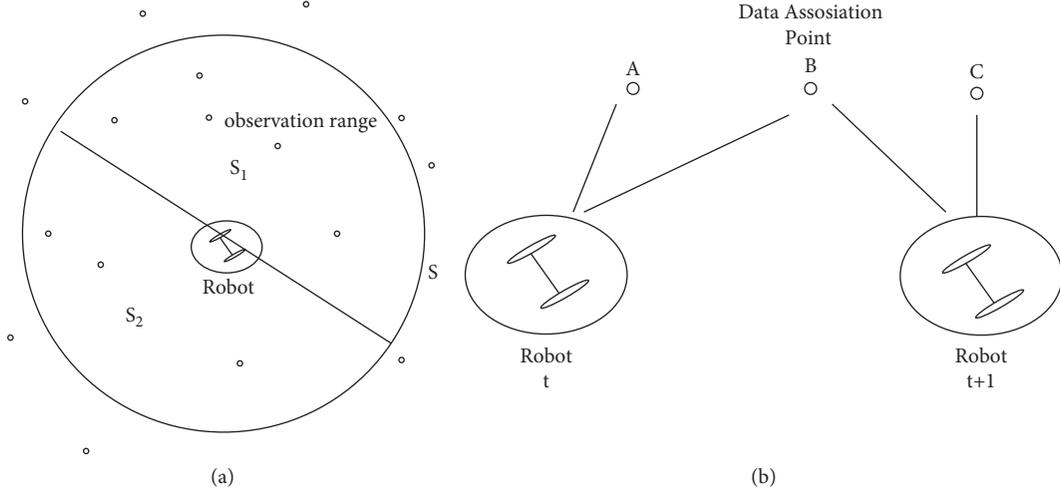


FIGURE 2: (a) Observation range; (b) local data association.

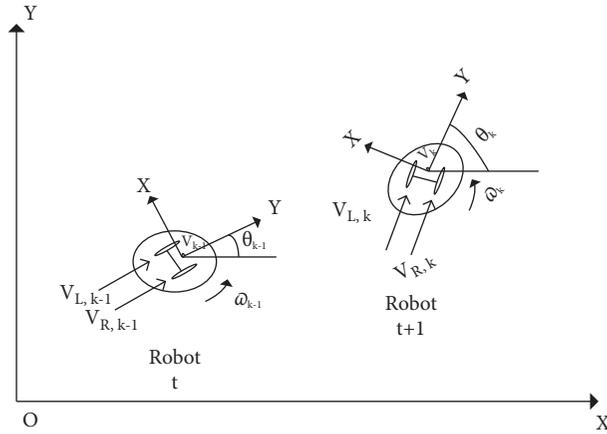


FIGURE 3: Model of mobile robot.

**7.1. Sensor Features Detection.** Sixteen Polaroid sonar sensors are individually with a beam width of  $30^\circ$ . The robot explored the home environment and was manually controlled at an average speed of 0.2 m/s. During this exploration phase, the sonar ring acquired range data at a frequency of 1 Hz (i.e., 16 data sets per 1 s).

By testing the SLAM performance of the improved RBPF, we get a successful experiment of robot localization and mapping. In Figure 4(a) and Figure 4(b), the black boxes are obstacles located by ultrasonic sensors and they have been mapped by improved RBPF. The blue trajectory is the robot motion localization; light green small arcs are the ultrasonic scanning signs. The average localization error has been compared with other filtering algorithms; with the iterations increase, the proposed improved RBPF has a faster convergence than others, as shown in Figure 5. The improved RBPF in the SLAM problem not only reduces the localization error but also has a faster convergence, because the GWO algorithm is used to assist in the resampling.

When applying the GWO algorithm to assist the resampling, randomly initialize artificial wolves in  $X_t^p$ , its

quantity  $n = 200$ , maximum iterations  $k_{\max} = 100$ , searching wolves scaling factor  $\alpha = 0.3$ , maximum number of searches  $T_{\max} = 20$ , distance determinant factor  $\omega = 50$ , step length factor  $S = 0.01$ , updating scaling factor  $\beta = 0.5$ ,  $a = 10$ , and  $\lambda = 20$ . The before and after resampling particles change trend is shown in Figure 6.

In Figure 6, a large swarm of red particles in degeneracy needs to be resampled, and the resampling method is the GWO; namely, remove the amount of  $N$  particles with small weights before resampling and producing the same amount of  $N$  particles in the vicinity of the approximate maximum weight of particle to replace the original ones after resampling. The new generation particles are represented by the black dots in Figure 6; the distribution of black dots does not have an ideal distribution but not far away from the maximum weight, since the GWO encircling wolves have a small distance from the leader wolf. Such distribution is enough to satisfy the SV value in a very small range and the change of SV curve shown in Figure 7, and all the results prove that GWO resampling has a good ID quality. The most important result is that GWO resampling not only ensures the diversity

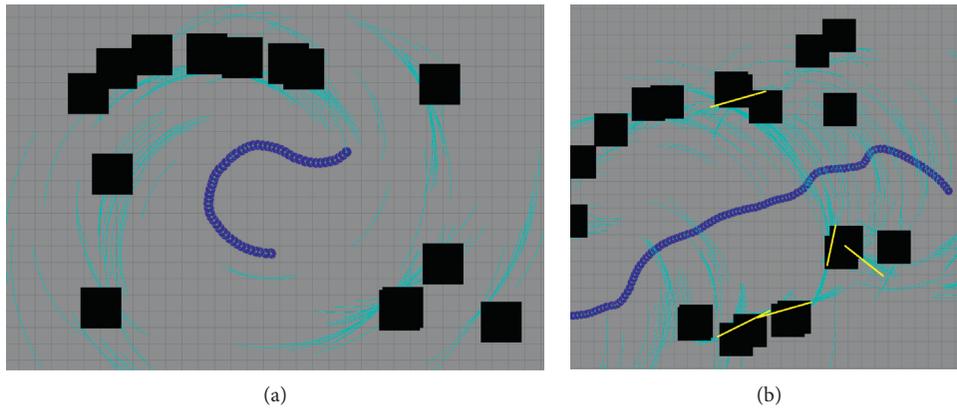


FIGURE 4: (a) SLAM experiment 1; (b) SLAM experiment 2.

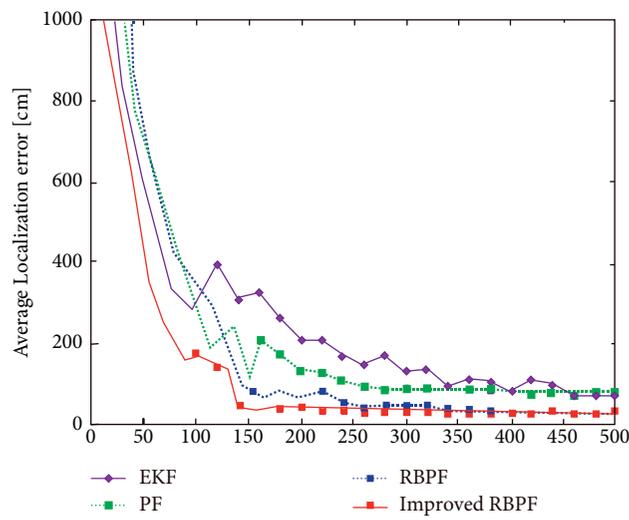


FIGURE 5: Average localization error with iterations.

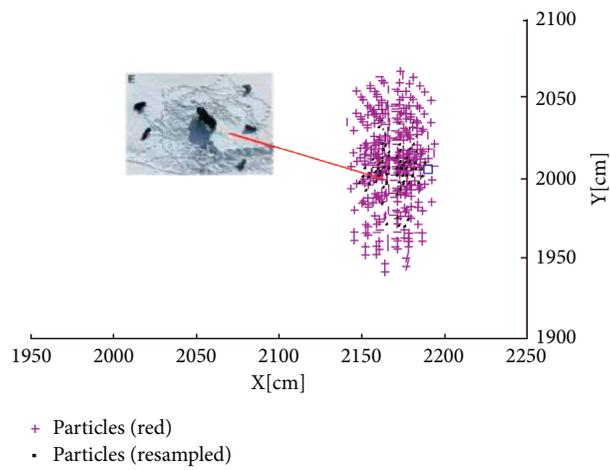


FIGURE 6: GWO-assisted resampling illustration.

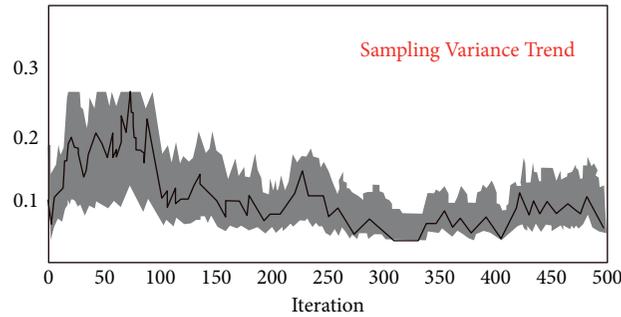


FIGURE 7: GWO resampling SV trend.

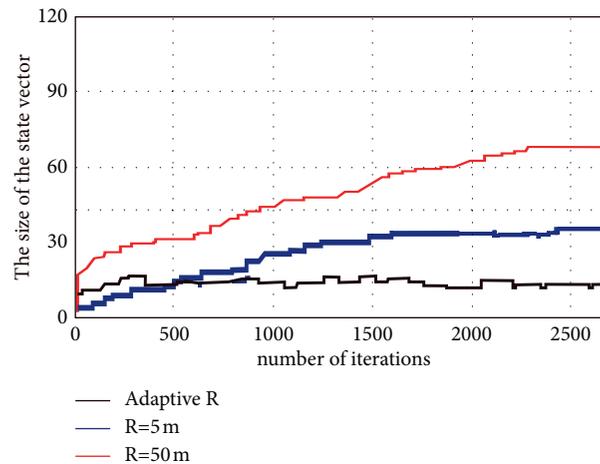


FIGURE 8: Size of the KF state vector.

of particles but also ensures the particles distribution approximate maximum weight after resampling.

This paper also proposes a new local data association method; experiments with the adjustment of domain of data association are made, as shown in Figure 8. We can see that the proposed method has a successful control of the length of state vector with KF prediction iteration, which simplifies the calculation of complexity and improves the efficiency. The most important thing is that the proposed local data association method in this paper can have a successfully accurate SLAM with localization error controlled in a certain range and also avoid the complexity of normal data association and its endless growth with more new landmarks.

## 8. Conclusion

To solve the SLAM problem of mobile robot, this paper adopts an improved RBPF filtering scheme; the important SIR is combined with an artificial intelligent algorithm GWO, which makes the improved RBPF with fast convergence. The most important thing is that the improved RBPF is able to avoid the normal resampling only considering the particles weight ID quality but ignoring the diversity of

particles having the significance of the localization of the robot by posterior probability distribution. In addition, this paper also improves the normal NN data association method and proposes a local data association-adaptive observation range (Range-SLAM) which not only reduces the complexity of the data association but also improves the efficiency of the NN algorithm as well as effectively avoiding the KF state vector from unlimited growth and reducing the computation complexity for RBPF-SLAM.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this study.

## Acknowledgments

The authors are grateful for the open-access datasets in GitHub. The research was funded by the 2021 High-level

Talents Research Support Program of Shenyang Ligong University, China (No. 1010147001017), and also by the Basic Research Project of the University of Science and Technology Liaoning, China.

## References

- [1] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics and Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.
- [2] S. Gagliano, F. Cairone, A. Amenta, and M. Bucolo, "A real time feed forward control of slug flow in microchannels," *Energies*, vol. 12, no. 13, Article ID 2556, 2019.
- [3] S. Gagliano, G. Stella, and A. M. Bucolo, "Real-time detection of slug velocity in microchannels," *Micromachines*, vol. 11, no. 3, Article ID 241, 2020.
- [4] F. Pei, M. Zhu, and X. Wu, "A decorrelated distributed EKF-SLAM system for the autonomous navigation of mobile robots," *Journal of Intelligent and Robotic Systems*, vol. 98, no. 3, pp. 819–829, 2020.
- [5] H. Lee, J. Chun, and K. Jeon, "Experimental results and posterior cramer-rao bound analysis of EKF-based radar SLAM with odometer bias compensation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 1, pp. 310–324, 2020.
- [6] J. S. Lee, S. Y. Nam, and W. K. Chung, "Robust RBPF-SLAM for indoor mobile robots using sonar sensors in non-static environments," *Advanced Robotics*, vol. 25, no. 9-10, pp. 1227–1248, 2011.
- [7] H. Jo, H. M. Cho, S. Jo, and E. Kim, "Efficient grid-based rao-blackwellized particle filter SLAM with interparticle map sharing," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 2, pp. 714–724, 2018.
- [8] T.-c. Li, G. Villarrubia, S.-D. Sun, J. M. Corchado, and J. Bajo, "Resampling methods for particle filtering: identical distribution, a new method, and comparable study," *Frontiers of Information Technology & Electronic Engineering*, vol. 16, no. 11, pp. 969–984, 2015.
- [9] T. Li, S. Sun, T. P. Sattar, and J. M. Corchado, "Fight sample degeneracy and impoverishment in particle filters: a review of intelligent approaches," *Expert Systems with Applications*, vol. 41, no. 8, pp. 3944–3954, 2014.
- [10] A. Doucet, N. De Freitas, K. Murphy, and S. J. Russell, "Rao-Blackwellised particle filtering for dynamic Bayesian networks," 2013, <https://arxiv.org/abs/1301.3853>.
- [11] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [12] J. Wang, L. Zhao, and X. Su, "Marginalized particle flow filter," *Circuits, Systems, and Signal Processing*, vol. 38, no. 7, pp. 3152–3169, 2019.
- [13] Newnes, *Swarm Intelligence and Bio-Inspired Computation: Theory and Applications*, Newnes, London, UK, 2013.
- [14] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [15] J. Zhang, L. Yuan, T. Ran, T. Qing, and H. Li, "Bayesian nonparametric object association for semantic SLAM," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5493–5500, 2021.
- [16] D. Liu, "A data association algorithm for SLAM based on central difference joint compatibility criterion and clustering," *Robotica*, vol. 39, no. 9, pp. 1674–1691, 2021.